

Instantaneous Quantum Channel Estimation during Quantum Information Processing

Yuichiro Fujiwara*

*Division of Physics, Mathematics and Astronomy,
California Institute of Technology, MC 253-37, Pasadena, California 91125, USA*
(Dated: May 27, 2014)

We present a nonintrusive method for reliably estimating the noise level during quantum computation and quantum communication protected by quantum error-correcting codes. As preprocessing of quantum error correction, our scheme estimates the current noise level through a negligible amount of classical computation using error syndromes and updates the decoder's knowledge on the spot before inferring the locations of errors. This preprocessing requires no additional quantum interaction or modification in the system. The estimate can be of higher quality than the maximum likelihood estimate based on perfect knowledge of channel parameters, thereby eliminating the need of the unrealistic assumption that the decoder accurately knows channel parameters a priori. Simulations demonstrate that not only can the decoder pick up on a change of channel parameters, but even if the channel stays the same, a quantum low-density parity-check code can perform better when the decoder exploits the on-the-spot estimates instead of the true error probabilities of the quantum channel.

PACS numbers: 03.67.Pp, 03.67.Lx, 03.67.Hk, 03.67.-a

I. INTRODUCTION

Protecting quantum information from noise is of paramount importance to quantum information processing because *qubits*, the information carriers, are fragile. *Quantum error-correcting codes* are schemes that encode quantum information into physical qubits in such a way that errors can be corrected [1].

Before implementing and performing quantum error correction, ideally we would like a very accurate channel model, that is, prior knowledge about how errors would manifest on qubits. The procedure of identifying the behavior of a noisy channel is called *channel estimation* or *parameter estimation* [2]. The known methods employ the technique called *quantum process tomography* [3], where probe qubits are fed to the channel and then the behavior of the channel is estimated from the outcome. In the context of fighting against noise, this standard approach allows for identifying what kind of error would occur and how frequently.

To correct any kind of error in a typical general channel model, one should be able to correct two types of errors, namely *bit flips* caused by Pauli operator X and *phase flips* caused by another kind of Pauli operator Z [4]. In a situation where the channel introduces a bit flip and/or phase flip on each qubit independently with certain probabilities, which we assume in most of this paper, properly implemented parameter estimation may quite accurately reveal the probability p_X that the channel introduces the X error on each qubit and the same statistical information p_Z about the Z error. In an ideal situation, one would gain the true values of p_X and p_Z , or *perfect knowledge* of the channel parameters.

When constructing a quantum information processing apparatus, we need to accurately estimate the channel parameters for each component so that we can install an appropriate quantum error-correcting code tailored to the identified channel behavior. However, even if we assume that channel parameters can be estimated with no error at the time of constructing an apparatus, there remain important problems in channel estimation for quantum error correction.

A trivial issue is that the known estimation methods can not be completed instantly on an apparatus that is currently operating. For instance, this means that any behavioral change in error pattern or frequency during quantum information processing can not be detected on the spot before attempting quantum error correction.

A subtler but equally critical problem is that perfect knowledge of channel parameters such as the true value of p_X is not the most helpful side information for quantum error correction once it goes into operation. In other words, while accurate knowledge of parameters is necessary to choose the right quantum error-correcting code, it is not the right information for realizing the full potential of the chosen quantum error-correcting code.

To make the latter point clearer, consider the following simple error correction for classical data transmission. The sender transmits binary information represented by 0's and 1's. Assume that the channel flips the symbol of each bit with probability, say, $p = \frac{1}{4}$. The simplest error correction scheme is to send the same symbol multiple times. Assume that the sender transmits 5 copies of each bit in a row so that 0 is sent as 00000 and 1 is encoded as 11111. If the receiver knows the error probability $p = \frac{1}{4} < \frac{1}{2}$, the most logical way to infer the correct symbol is by majority vote. For instance, if the received message is 01000, the correct message is most likely 0.

Is the true value of the error probability p the most useful information to this simple error-correcting code? The answer is no. Excluding the actual error locations

* yuichiro.fujiwara@caltech.edu

which are assumed to be unknown, it is the actual number of errors that is most useful. If the number of errors is less than or equal to 2, both knowing p and knowing the number of errors will lead to the correct guess. If there are more errors, however, the most logical inference based on perfect knowledge of p fails to reach the correct message. Yet, knowing the actual number of errors always correctly reveals the original message. For instance, if the receiver is told that 3 bits are flipped when 11010 is received, the most logical inference is 00000, which is trivially correct.

In general, from the viewpoint of the receiver, the actual number of erroneous bits, or equivalently the *current noise level*, is more useful side information than perfect knowledge of error probability p because the channel parameter p only tells what the noise level is on average. Therefore, it is natural to ask whether quantum error correction can also exploit knowledge of the current noise level and, if positive, whether it is possible to estimate it on the spot before inferring errors during quantum information processing. This paper answers both questions in the affirmative.

We show that it is possible to reliably estimate the number of errors on encoded qubits without disturbing the quantum state in such a way that no additional quantum circuit or quantum interaction is required. The estimate is obtained instantaneously through a negligible amount of classical computation before the decoder of the quantum error-correcting code starts to infer the types and locations of errors. In other words, the current noise level can be estimated as preprocessing of quantum error correction at virtually no cost.

The estimate can be fed into the decoder each time to make quantum error correction more reliable by letting it adaptively respond to the current noise level. This means that not only can the decoder pick up on a change of a channel parameter, but even if p_X and p_Z stay exactly the same, it can also follow the natural deviations from the expected number of errors to the extent the accuracy and precision of estimation allows.

It is shown that our instantaneous quantum channel estimation can be implemented with a quantum error-correcting code that can take advantage of the *sum-product algorithm* [5, 6], which is among the most sophisticated and popular decoding methods available in coding theory. Simulations demonstrate that the on-the-spot estimate can be of very high quality to the extent that the decoder no longer needs perfect knowledge of channel parameters during quantum error correction.

II. PRELIMINARIES

Our estimation scheme is integrated with quantum error correction that takes advantage of classical error correction. For the basic notions and facts in classical coding theory and quantum error correction, we refer the reader to standard textbooks such as [1, 4, 5].

A *binary linear* $[n, k]$ code is a k -dimensional subspace \mathcal{C} of the n -dimensional vector space \mathbb{F}_2^n over the finite field \mathbb{F}_2 with exactly two elements $\{0, 1\}$, so that \mathcal{C} encodes k bits of information into n physical bits as a classical error-correcting code. The *dual code* \mathcal{C}^\perp of \mathcal{C} is defined as $\mathcal{C}^\perp = \{\mathbf{d} \in \mathbb{F}_2^n \mid \mathbf{c} \cdot \mathbf{d} = \mathbf{0} \text{ for any } \mathbf{c} \in \mathcal{C}\}$.

A binary linear code \mathcal{C} can be seen as the null space $\{\mathbf{c} \in \mathbb{F}_2^n \mid H\mathbf{c} = \mathbf{0}\}$ of some $(n - k) \times n$ matrix H over \mathbb{F}_2 . The matrix H is called a *parity-check matrix* of \mathcal{C} . Take an n -dimensional vector $\mathbf{e} = (e_0, \dots, e_{n-1}) \in \mathbb{F}_2^n$. Assume that a message $\mathbf{c} = (c_0, \dots, c_{n-1}) \in \mathcal{C}$ is sent and that the vector $\mathbf{c} + \mathbf{e}$ is received, which means that the bit c_i is flipped by the channel if $e_i = 1$ and it is intact if $e_i = 0$. The traditional error correction method computes the k -dimensional vector $\mathbf{s} = H(\mathbf{c} + \mathbf{e}) = H\mathbf{e}$, called the *syndrome*, and then infers \mathbf{e} from \mathbf{s} with the help of side information such as the error probability p .

Similar to the classical case, an $[[n, k]]$ quantum error-correcting code encodes k qubits of quantum information into n physical qubits. A quantum error-correcting code can be constructed from binary linear codes. Let $\mathcal{C}_1, \mathcal{C}_2$ be a pair of binary linear codes of parameters $[n, k_1]$ and $[n, k_2]$ respectively. If \mathcal{C}_1 contains the dual code \mathcal{C}_2^\perp , that is, $\mathcal{C}_2^\perp \subseteq \mathcal{C}_1$, then an $[[n, k_1 + k_2 - n]]$ quantum error-correcting code, called a *Calderbank-Shor-Steane* (CSS) code [7, 8], can be constructed.

For a unitary operator U and a binary vector $\mathbf{a} = (a_0, \dots, a_{n-1}) \in \mathbb{F}_2^n$, define $U^{\mathbf{a}}$ as the n -fold tensor product $O_0 \otimes \dots \otimes O_{n-1}$, where $O_i = U$ if $a_i = 1$ and O_i is the identity operator otherwise. A CSS code exploits a technique called *discretization* so that its error detection measurement takes any error operator E introduced by the channel to a combination of bit flips X , phase flips Z , and both at the same time. Let H_1 and H_2 be parity-check matrices of the binary linear codes \mathcal{C}_1 and \mathcal{C}_2 such that $\mathcal{C}_2^\perp \subseteq \mathcal{C}_1$. Then, with $2k$ ancilla qubits, appropriate measurement discretizes the error E on an n -qubit state $|\psi\rangle$ encoded by a CSS code as follows:

$$E|\psi\rangle \rightarrow |H_1\mathbf{e}_X\rangle |H_2\mathbf{e}_Z\rangle X^{\mathbf{e}_X} Z^{\mathbf{e}_Z} |\psi\rangle,$$

where $\mathbf{e}_X = (e_0^X, \dots, e_{n-1}^X) \in \mathbb{F}_2^n$ is the n -dimensional vector such that $e_i^X = 1$ if a bit flip occurred on the i th qubit and $e_i^X = 0$ otherwise, and \mathbf{e}_Z is the n -dimensional vector representing phase flips the same way. Measuring ancilla qubits, we obtain the syndrome $H_1\mathbf{e}_X$ for bit flips as a binary $(n - k_1)$ -dimensional vector and the other syndrome $H_2\mathbf{e}_Z$ for phase flips as a binary $(n - k_2)$ -dimensional vector. By exploiting the error correction method for binary linear codes, we may infer \mathbf{e}_X and \mathbf{e}_Z from the syndromes and side information such as p_X and p_Z we learned when constructing the apparatus.

What we prove here is that if H_1 and H_2 are chosen suitably, a tiny amount of classical computation with the syndromes can estimate the current noise level, or equivalently the numbers of bit flips and phase flips. Thus, the next step where the decoder infers \mathbf{e}_X and \mathbf{e}_Z can exploit the estimated current noise level as more useful and updated side information.

III. INSTANTANEOUS NOISE LEVEL ESTIMATION

Now we show how to estimate the current noise level for bit flips from $H_1 \mathbf{e}_X$ before inferring the error vector \mathbf{e}_X . Because the same estimation can be performed for phase flips from $H_1 \mathbf{e}_Z$, the rest of this paper focuses on bit flips. A remark on extensions to other channel models will be given at the end.

We assume that the X operator acts on each qubit independently with probability $p < \frac{1}{2}$. Hence, in terms of bit flips, the quantum channel is modeled as a binary symmetric channel with error probability p in the language of coding theory. The current error probability p is not necessarily equal to the original value p_X because we assume that the frequency of errors may change. No prior knowledge about the current value is assumed except the assumption that it is less than $\frac{1}{2}$.

Recall that H_1 is a parity-check matrix of binary linear $[n, k_1]$ code \mathcal{C}_1 with $n - k_1$ linearly independent rows and n columns. The *weight* $\text{wt}(\mathbf{r})$ of a vector \mathbf{r} over \mathbb{F}_2 is the number of nonzero entries, that is, the number of 1's. For the sake of simplicity, we assume that every row of H_1 is of weight r .

Our estimation bases on the following approximation.

Proposition 1 ([9]) *Let \mathbf{e}_X be a Bernoulli process with n trials and probability p , and H_1 an $(n - k_1) \times n$ parity-check matrix of a binary linear $[n, k_1]$ code in which every row is of weight r . The weight of the syndrome $\mathbf{s}_1 = H_1 \mathbf{e}_X$ can be approximated by a random variable that follows the binomial distribution of parameters $n - k_1$ and $q_{r,p}$, where*

$$q_{r,p} = \sum_{\substack{1 \leq i \leq r \\ i \text{ odd}}} \binom{r}{i} p^i (1-p)^{r-i}. \quad (1)$$

Note that the right-hand side of Equation (1) can be simplified to $\frac{1-(1-2p)^r}{2}$. Assuming the approximation given in Proposition 1, the probability $P[\text{wt}(\mathbf{s}_1) = s]$ that the weight of the syndrome is s is

$$P[\text{wt}(\mathbf{s}_1) = s] = \binom{n - k_1}{s} q_{r,p}^s (1 - q_{r,p})^{n - k_1 - s}.$$

Hence, given that $\text{wt}(\mathbf{s}_1) = s$, the maximum likelihood estimate \hat{p}_s of p is

$$\begin{aligned} \hat{p}_s &= \arg \max_x \{ q_{r,x}^s (1 - q_{r,x})^{n - k_1 - s} \} \\ &= \begin{cases} \frac{1}{2} - \frac{1}{2} \left(1 - \frac{2s}{n - k_1} \right)^{\frac{1}{r}} & \text{if } \frac{s}{n - k_1} \leq \frac{1}{2}, \\ \frac{1}{2} & \text{otherwise.} \end{cases} \quad (2) \end{aligned}$$

The point is that the nearest integer $[n\hat{p}_s]$ is an estimate of the number of bit flips that actually occurred within the encoded n -qubit block because \hat{p}_s is calculated from the Bernoulli process \mathbf{e}_X of a single whole set of n trials.

In other words, \hat{p}_s is more strongly correlated with what just happened on qubits at hand than how the channel behavior would average out over the course of time.

Another important fact is that our estimation scheme does not impose any overhead except the negligibly small calculation by the closed form given in Equation (2). This is because the only necessary information, which is the syndromes, is required regardless by the inference of the types and locations of errors.

The mean μ of $\hat{p}_{\text{wt}(\mathbf{s}_1)}$ is

$$\mu = \frac{1}{2} - \frac{1}{2} \sum_{i=0}^{\lfloor \frac{m}{2} \rfloor} \binom{m}{i} q_{r,x}^i (1 - q_{r,x})^{m-i} \left(1 - \frac{2i}{m} \right)^{\frac{1}{r}},$$

where $m = n - k_1$. When seen as an estimator of p , the mean squared error $\text{MSE}(\hat{p}_{\text{wt}(\mathbf{s}_1)})$ is

$$\begin{aligned} \text{MSE}(\hat{p}_{\text{wt}(\mathbf{s}_1)}) &= \mathbb{E} \left[(\hat{p}_{\text{wt}(\mathbf{s}_1)} - p)^2 \right] \\ &= p^2 - 2p\mu + \frac{1}{4} \\ &\quad + \frac{1}{4} \sum_{i=0}^{\lfloor \frac{m}{2} \rfloor} \binom{m}{i} q_{r,x}^i (1 - q_{r,x})^{m-i} \\ &\quad \times \left(\left(1 - \frac{2i}{m} \right)^{\frac{2}{r}} - 2 \left(1 - \frac{2i}{m} \right)^{\frac{1}{r}} \right). \end{aligned}$$

In principle, one may exactly compute the more relevant quantity, namely the expected discrepancy between a realization of $\frac{\text{wt}(\mathbf{e}_X)}{n}$ and our estimate, although it may be computationally infeasible for a parity-check matrix of practical size.

The key to effectively exploiting our scheme is choosing a suitable parity-check matrix such that the approximation in Proposition 1 is reasonable and such that the estimation is of high quality. It should be noted that it is not the same as choosing a suitable binary linear code because one same code has many different parity-check matrices of different column and row weights.

In general, making column weights smaller decreases correlations between bits in a syndrome and hence improves the accuracy of the approximation in Proposition 1. By the same token, the number of overlaps of the positions of 1's between any pair of rows should be small. In addition, the row weight r should be relatively small to keep $q_{r,p}$ sensitive to p . Finally, all else being equal, larger n and smaller k_1 are desirable because estimation becomes more reliable as $m = n - k_1$ becomes larger.

Here we illustrate our estimation method through an example case. The quantum error-correcting code we use exploits a state-of-the-art decoding technique, the sum-product algorithm, for binary linear codes. A *low-density parity-check* (LDPC) code is a linear code that admits a parity-check matrix with a small number of nonzero entries such that iterative decoding algorithms perform well [5]. It is known that well-designed LDPC codes can nearly attain the channel capacity, which is the theoretical limit of error correction [10]. A CSS code that

uses LDPC codes as its ingredients is called a *quantum LDPC code*. Typically, the column weights of a parity-check matrix of an LDPC code are only a few to several. Row weights are also quite small. A well-designed LDPC code tends to have a very small number of overlaps of the positions of 1's between a pair of rows because rows with more than one overlap degrades the performance of its decoding algorithm. In addition, the sum-product algorithm and most of its variations achieve their characteristic excellent error correction performance by exploiting information about the noise level (see [11–14] for the effect of a mismatch between actual and assumed noise levels). Therefore, quantum LDPC codes with good error correction performance are naturally suited for our purpose.

Among various known construction techniques for quantum LDPC codes, Construction B given in [6] is one of the most successful ones. Following Figure 6 of [6], we set $n = 3786$ and $k_1 = 2366$. The row and column weights significantly affect the expected performance of an LDPC code of given parameters n and k_1 [15]. We adjusted our parity-check matrix so that the block error rate (BLER) reaches 5×10^{-5} roughly at $p = 0.02$ with the sum-product algorithm. Every row is of weight 24. The column weights are nearly uniform with the mean weight being 10. This LDPC code contains its dual code, so that the resulting quantum LDPC code is of parameters [[3786, 946]].

We simulated X errors by binary symmetric channels and estimated the current noise level each time by Equation (2). Table I shows the mean of the squared errors of estimates obtained through simulations. Note that the maximum likelihood estimate of the current noise level by perfect knowledge of p is p itself. Hence, the corresponding quality measure for this perfect knowledge estimator is the variance of the binomial distribution divided by n^2 . As shown in Table I, our estimates are of higher quality than the maximum likelihood estimates based on perfect knowledge in terms of expected discrepancy.

TABLE I. Quality of estimates.

p	$\text{MSE}(\hat{p})$	$\text{MSE}(p)$	$n\hat{p}/\text{wt}(\mathbf{e}_X)$
0.0175	1.0×10^{-6}	4.5×10^{-6}	1.007
0.02	1.4×10^{-6}	5.1×10^{-6}	1.008
0.0225	2.0×10^{-6}	5.8×10^{-6}	1.008
0.025	2.8×10^{-6}	6.4×10^{-6}	1.009
0.0275	3.8×10^{-6}	7.0×10^{-6}	1.009
0.03	5.1×10^{-6}	7.6×10^{-6}	1.010
0.0325	6.9×10^{-6}	8.3×10^{-6}	1.011

Figure 1 plots the BLER b_p of the LDPC code decoded by the sum-product algorithm over a binary symmetric channel with error probability p . If bit flips and phase flips are treated separately, the BLER of the corresponding quantum LDPC code achieves $1 - (1 - b_p)^2 \approx 2b_p$ over a depolarizing channel with equal error probability $\frac{p}{2}$ for each of the three types of Pauli errors. We compared

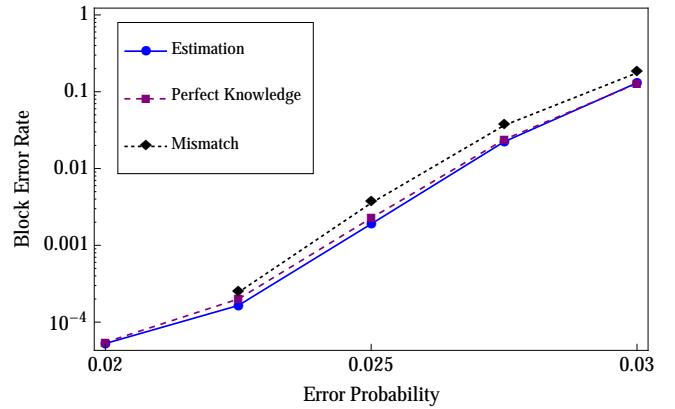


FIG. 1. (Color online) Performance of quantum LDPC code.

three scenarios: the decoder always assumes $p = 0.02$ regardless of the actual channel parameter p , which is the most realistic assumption without our method; the decoder has perfect knowledge of p , which is unrealistically optimistic; and the decoder uses the estimate \hat{p}_s each time. As illustrated in Figure 1, feeding \hat{p}_s to the sum-product decoder completely suppresses the detrimental effect of the mismatch between actual and assumed noise levels. In fact, decoding with the estimate \hat{p}_s achieved a better BLER for $p < 0.03$ than with the true value of p because \hat{p}_s is more strongly correlated with the actual number of bit flips than with what is expected on average. The confidence level of the detection of improvement over perfect knowledge at $p = 0.0225$ is 3.9σ . No advantage over perfect knowledge was observed when the quantum LDPC code was overwhelmed by too much noise or overkill for too low a noise level.

IV. CONCLUDING REMARKS

The instantaneous quantum channel estimation we developed here makes the unrealistic, ideal assumption of perfect knowledge unnecessary when appropriate quantum error-correcting codes and their sophisticated decoding algorithms are employed. In fact, not only does our scheme suppress the negative effect of incorrect knowledge of channel parameters, but it can also give a better BLER than if perfect knowledge is available because to the eye of the decoder, the actual number of errors is more relevant than how the channel behaves on average.

It should be noted that the idea presented here is particularly more suited for quantum error correction than classical error correction. This is because LDPC codes in electrical engineering typically make use of soft information represented by continuous values instead of binary syndromes. Since good nonintrusive estimation can be done using soft information in the classical case [16], it is not as appealing if there is no other reason to convert soft information into binary data, a process known as a *hard decision* [17]. Contrary to the classical case, active

quantum error correction naturally involves discretization, which is a form of hard decision. Hence, quantum information processing is exactly the kind of application that benefits from our method.

It is possible to extend our method to different channel models. For instance, decoherence due to amplitude and phase damping can be approximated by Pauli operators X , Y , and Z through Pauli twirling [18]. In this case, we can derive the current noise level for Y from our estimates

of p_X and p_Z . Hence, the decoder can also exploit the correlation between bit flips and phase flips due to the Y operator [6, 19, 20].

An interesting question is how to effectively use knowledge of the current noise level. While we simply used the estimation for the initialization of the sum-product algorithm by feeding the estimate as the assumed noise level, there may be a more sophisticated way to exploit the knowledge.

-
- [1] D. A. Lidar and T. A. Brun, eds., *Quantum Error Correction* (Cambridge Univ. Press, New York, 2013).
 - [2] A. Fujiwara, Phys. Rev. A **63**, 042304 (2001).
 - [3] M. Mohseni, A. T. Rezakhani, and D. A. Lidar, Phys. Rev. A **77**, 032322 (2008).
 - [4] M. A. Nielsen and I. L. Chuang, *Quantum Computation and Quantum Information* (Cambridge Univ. Press, New York, 2000).
 - [5] D. J. C. MacKay, *Information Theory, Inference, and Learning Algorithms* (Cambridge University Press, Cambridge, 2003).
 - [6] D. J. C. MacKay, G. Mitchison, and P. L. McFadden, IEEE Trans. Inf. Theory **50**, 2315 (2004).
 - [7] A. R. Calderbank and P. W. Shor, Phys. Rev. A **54**, 1098 (1996).
 - [8] A. M. Steane, Phys. Rev. Lett. **77**, 793 (1996).
 - [9] V. Toto-Zaraso, A. Roumy, and C. Guillemot, IEEE Commun. Lett. **15**, 232 (2011).
 - [10] N. Bonello, S. Chen, and L. Hanzo, IEEE Commun. Surveys Tutorials **13**, 3 (2011).
 - [11] D. J. C. MacKay and C. P. Hesketh, Electron. Notes Theoretical Comput. Sci. **74**, 1 (2003).
 - [12] L. Qi, G. Chen, C. Huijuan, and T. Kun, in *2006 IMACS Multiconference on Computational Engineering in Systems Applications* (Beijing, China, 2006) pp. 1600–1604.
 - [13] H. Saeedi and A. H. Banihashemi, IEEE Trans. Commun. **55**, 83 (2007).
 - [14] M. Hagiwara, M. P. C. Fossorier, and H. Imai, IEEE Trans. Inf. Theory **58**, 2321 (2012).
 - [15] T. J. Richardson and R. L. Urbanke, *Modern Coding Theory* (Cambridge Univ. Press, New York, 2008).
 - [16] D. R. Pauluzzi and N. C. Beaulieu, IEEE Trans. Commun. **48**, 1681 (2000).
 - [17] G. Lechner and C. Pacher, IEEE Commun. Lett. **17**, 2148 (2013).
 - [18] M. Silva, E. Magesan, D. W. Kribs, and J. Emerson, Phys. Rev. A **78**, 012347 (2008).
 - [19] Y.-J. Wang, B. C. Sanders, B.-M. Bai, and X.-M. Wang, IEEE Trans. Inf. Theory **58**, 1231 (2012).
 - [20] M. Denise, J. P. Tillich, and I. Andriyanova, in *Proc. IEEE Int. Symp. Inf. Theory* (2013) pp. 907–911.